Distributed Exercise Management (DEM) for HLA Simulations

Authors:
Schow, Greg, STRICOM
Freeman, Rhonda, TASC
Tony Lashley, TASC
John Pfalz, TASC
Swauger, Jeff, TASC DIS SEI Center
Lander, Brent, SAIC
12443 Research Parkway
Orlando, FL 32826

KEYWORDS
Exercise Management

ABSTRACT

The purpose of this paper is to report on the design approach and experiences of the Distributed Exercise Management (DEM) portion of the DISECT program. The Distributed Exercise Manager (DEM) provides exercise control and monitoring functions for large High Level Architecture (HLA) based simulation exercises. The DEM provides both a centralized simulation exercise monitoring and control capability, referred to as DEM Central, and distributed monitoring within the network architecture using distributed platforms located at each LAN, referred to as DEMvices. Exercise Control allows an exercise manager to initialize and execute a federation within the High Level Architecture (HLA) simulation environment. Run Time Infrastucture (RTI) Monitoring provides information about the operation of the federates in an exercise, as well as information on the RTI itself. Network Monitoring allows the status and performance of the network assets used for the simulation exercise to be evaluated and the network monitored for overload or error conditions. Work Station Load Monitoring allows the load of the individual computers supporting the exercise, from environment servers to simulators generating entities and interactions, to be monitored and evaluated.

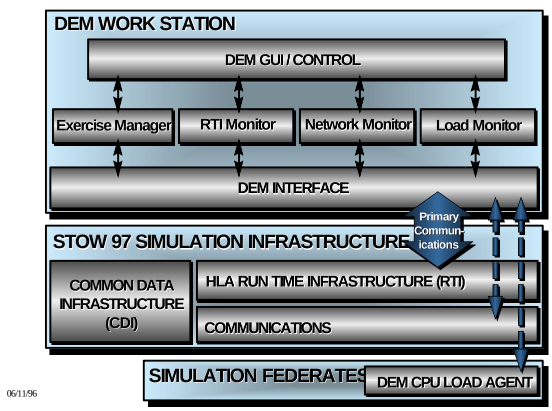
1.0 Introduction

The DISECT Distributed Exercise Manager (DEM) provides exercise control and monitoring functions for large High Level Architecture (HLA) based simulation exercises, such as STOW 97. Exercise monitoring functions provide an efficient method of monitoring the performance of the hardware and software infrastructure of the simulation exercise and of detecting and isolating any problems that arise during the exercise. Exercise control functions allow an exercise manager to start, stop, and otherwise direct the exercise.

The DEM provides four main functions; Exercise Control, RTI Monitoring, Network Monitoring, and Work Station (W/S) Load Monitoring. This provides both a centralized simulation exercise monitoring with control capability, referred to as DEM Central, and distributed monitoring within the network architecture using distributed platforms located at each LAN, referred to as DEMvices. Exercise Control allows an exercise manager to initialize and execute a federation within the High Level Architecture (HLA) simulation environment. RTI Monitoring provides information about the operation of the federates in an exercise, as well as information on the RTI itself. Network Monitoring allows the status and performance of the network assets used for the simulation exercise to be evaluated and the network monitored for overload or error conditions. Work Station Load Monitoring allows the load of the individual computers supporting the exercise, from environment servers to simulators generating entities and interactions, to be monitored and evaluated. Both Network Monitoring and Work Station Load Monitoring are necessary precursors to the capability of performing real time load balancing for distributed simulation exercises.

Many of the functions of DEM require corresponding capabilities and services to be provided by the RTI and federate software. RTI monitoring and exercise monitoring require that the RTI implemented for STOW 97 provide the capabilities described above. Network and workstation load monitoring are being implemented. The version of ModSAF being used in the STOW 97 exercise has been modified to incorporate the capability for DEM to accurately monitor the load of the ModSAF stations. A method exists using standard UNIX calls to provide at least some level of monitoring of systems that are not ModSAF based (e.g., environment servers, C4I systems). Several approaches to the task of network monitoring have been evaluated, and Simple Polling has been chosen as the approach that provides the best technical performance, cost, and risk.

Figure 1 - DEM Functional Block Diagram



2.0 Exercise Control

The goal of the Exercise Control portion of the DEM is to provide all of the functionality required to allow an exercise manager to initialize and execute a federation within the High Level Architecture (HLA) simulation environment. Initially, the following functions were proposed for implementation:

1. Federation Creation
2. Federation Destruction
3. Federation Pause
4. Federation Resume
5. Federation Save
6. Federation Restore

The RTI must provide methods which correspond to each of the aforementioned federation management functions in order for the DEM to supply this functionality to the exercise manager. DEM is to have the capability of dismissing a federate which has, for whatever reason, become unresponsive and unable to dismiss itself from the federation. The current functionality being proposed for the exercise control portion of the DEM is as follows:

1. Federation Creation
2. Federation Destruction
3. Federation Save
4. Federation Restore
5. Federation Pause
6. Federation Resume
7. Dismiss Federate

The RTI must still provide methods which correspond to each of the above mentioned functions in order for the DEM to supply this functionality to the exercise manager. Significant coordination is ongoing between TASC, SAIC, and Lincoln Labs to ensure this functionality is present for the STOW 97 exercise.

A graphical user interface is being designed to allow a simulation manager to execute these RTI methods. Within the HLA environment, it is the responsibility of the federates to honor requests from the RTI relating to these federation management functions. Therefore, correct operation of the DEM with regard to federation management requires each participating federate to implement these federation functions.

The exercise control functionality of the DEM will exist in only one place; DEM Central. The DEMvices on each STOW 97 WAN node, or LAN tail circuit will not have any exercise control capability. The rationale for this is that it is undesirable to have exercise control capability in more than one place. If there were multiple exercise control functions making requests of the RTI, confusion and conflicts could arise which would negatively effect the exercise. One exercise controller may want to stop federation execution, another may want to start or pause, all at the same time. These main exercise control functions should be manipulated in only one place. Therefore, exercise control is only allowed at DEM Central.

3.0 Run-Time Infrastructure (RTI) Monitor:

To ensure the proper execution of a federation within the HLA environment, an RTI Monitor is required. The latest release of the STOW 97 RTI has very little functionality built in to allow monitoring of

RTI and federation status. DEM requires certain functionality of the RTI to meet the requirements bestowed on it for STOW 97. This functionality will be implemented using the developing Management object Model (MOM) interface. Eight new functions are needed to meet these requirements and are described:

- **Report Federation Membership -** Ability to monitor which federates have joined the federation.
- **Report Federate Object Publication -** Ability to inspect which object classes are being published by any federate within the federation.
- **Report Federate Object Subscription -** Ability to inspect which object classes are being subscribed to by any federate within the federation.
- **Report Federate Attribute Publication -** Ability to inspect any attribute published by any federate within the federation.
- **Report Federate Attribute Subscription -** Ability to inspect any attribute subscribed to by any federate within the federation.
- **Report Federate Interaction Published -** Ability to inspect which interactions are being published by any federate within the federation.
- **Report Federate Interaction Subscription -** Ability to inspect which interactions are being subscribed to by any federate within the federation.
- **RTI Status Report -** Ability to request performance metrics available on the current state of the RTI and the federation.

The following paragraphs explain these functions in more detail, and list some examples of how they would be used in an HLA exercise:

Report Federation Membership

This functionality is extremely critical to the RTI monitoring. It will allow the user to monitor which federates have joined the federation, expected federates which have not joined, and federates which have joined but were not expected. With this capability an exercise manager has much greater insight as to which players are ready to start federation execution before the start of an exercise.

Example: Upon creation of the STOW 97 federation, the DEM RTI Monitor will display a list of federates from a pre-defined "STOW 97 federation" configuration file. Each time the "Report Federation Membership" function is invoked, the federation list is updated, and each federate name is color coded to depict its present status (joined / not joined). The exercise manager can monitor federates as they join the STOW 97 federation, and announce StartEx when satisfied that all required federates have joined and are ready to start the exercise. The secondary purpose of this feature is to display federates which have joined but were not scheduled to be a part of the federation. These federates will be displayed in a color that expresses a warning to the user. The exercise manager can monitor these "unexpected" federates, and determine whether to run the exercise with them, or make them leave the federation before starting the exercise.

Report Federate Object Published

This functionality will allow the RTI monitor to request from the RTI all the different objects that one particular federate is publishing. This functionality is important to verify interoperability with the Federation Object Model (FOM) and other federates in the federation. With this capability, the RTI Monitor will be able to verify the objects according to the FOM and raise a flag if a federate is publishing objects that are not compatible. This capability will also lend itself as a tool used in integration and testing to verify what data is being published.

Example: If a federate is supposed to be publishing an object that is a T72 and a different federate displays that entity as a T80, this functionality will give the RTI monitor the ability to examine the object for the appropriate entity type.

Report Federate Object Subscription

This functionality will allow the RTI monitor to request from the RTI, all the different objects one particular federate is subscribing to. This functionality is important to verify interoperability with the Federation Object Model (FOM) and other federates in the federation. With this capability, the RTI Monitor will be able to verify that the federate is subscribing to the proper objects according to the FOM . This capability will also lend itself as a tool used in integration and testing to verify what data is being subscribed to by each federate.

Example: If a federate is supposed to be subscribing to objects of a type "M1" but is displaying no M1's when there is actually an M1; this functionality will verify what object the federate is actually subscribing to.

Report Federate Attribute Published

This functionality will allow the RTI monitor to request from the RTI, all the different attributes one particular federate is publishing. This functionality is important to verify interoperability with the Federation Object Model (FOM) and other federates in the federation. With this capability, the RTI Monitor will be able to verify the attributes according to the FOM and raise a flag if a federate is publishing attributes that are not compatible. This capability will also lend itself as a tool used in integration and testing to verify what data is being published.

Example: If a federate is displayed on a PVD with it's hull orientation facing opposite to the rest of the company, the RTI monitor could examine the Attributes that the federate is publishing and verify where the entity orientation is facing.

Report Federate Attribute Subscription

This functionality will allow the RTI monitor to request from the RTI, all the different attributes one particular federate is subscribing to. This functionality is important to verify interoperability with the Federation Object Model (FOM) and other federates in the federation. With this capability, the RTI Monitor will be able to verify that the federate is subscribing to the correct attribute according to the FOM . This capability will also lend itself as a tool used in integration and testing to verify what data is being subscribed to by each federate.

Example: If a federate is displaying another federate and the displayed federate was moving but the hull orientation was 90 degrees to the direction of movement, the RTI Monitor could determine if the federate was subscribing to the correct attributes.

Report Federate Interaction Published

This functionality will allow the RTI monitor to request from the RTI, all the different interactions one particular federate is publishing. This functionality is important to verify interoperability with the Federation Object Model (FOM) and other federates in the federation. With this capability, the RTI Monitor will be able to verify the interactions according to the FOM and raise a flag if a federate is publishing interactions that are not compatible. This capability will also lend itself as a tool used in integration and testing to verify what data is being published.

Example: If a federate received multiple hits but never showed any damage from another federates view, the RTI Monitor could monitor the federate receiving firing and verify if any detonation interactions are being sent and what information is contained in the interaction.

Report Federate Interaction Subscription

This functionality will allow the RTI monitor to request from the RTI, all the different interactions one particular federate is subscribing to. This functionality is important to verify interoperability with the Federation Object Model (FOM) and other federates in the federation. With this capability, the RTI Monitor will be able to verify that the federate is subscribing to the correct interactions according to the FOM . This capability will also lend itself as a tool used in integration and testing to verify what data is being subscribed to by each federate.

Example: If a federate received direct fire and is displaying damage but other federates are not showing any damage, the RTI Monitor could evaluate what interaction the federates that are not showing any damage are subscribing to and verify that they are subscribing to the appropriate interactions.

RTI Status Report

This functionality will allow the RTI monitor to request from the RTI a status report on all statistical metrics available on the current status of the RTI. Metrics which could be tracked could include latency (updates / updates received), general counts (# federates), RTI busyness, bandwidth usage and bandwidth available.

Example: When federates are not receiving updates in a timely manner, the RTI Monitor will have the capability to poll the RTI. The data received will give the user information to assist in determining if there is a problem with the RTI, federation execution, or certain federates.

It is important to recall that these functions are needed in order to meet STOW 97 requirements for RTI Monitoring. Without this functionality, little or no status of the RTI will be available. It is possible obtain a small amount of RTI/federation status without this functionality by using a promiscuous federate, however, this could create a bandwidth bottleneck since this federate would be required to subscribe to all objects, interactions, and attributes. Even then, the information received would be small when compared with the amount available using the new functions.

4.0 Network and Workstation (W/S) Load Monitoring

Monitoring of the network and work station loading are very closely related. The same mechanisms can be used to accomplish both of these functions. As a result of their close interrelationship, these two functions are discussed and documented together in this section. The following paragraphs discuss the specific goals for network and W/S load monitoring, and describe the approach chosen to provide these functions, and the supporting rationale.

Specific goals for Network Monitoring are to supply LAN managers with local network traffic conditions (average, peak and variance packets/sec for the LAN and individual Work Stations) and LAN to LAN connectivity and latency information. This monitoring must be done with minimal network bandwidth impact and is required to be an accurate measure of network traffic (minimal lost packet information). Impact to W/S CPU Utilization must also be minimized. The system is capable of identifying when specified limits are exceeded and report any problems to the local LAN manager, as well as the DEM Central operator. Information collected at local LANs should be generally available to the DEM Central operator. It should be pointed out that none of these approaches specifically addresses the issue of someone corrupting the STOW 97 network with packets that are not exercise related (e.g. ftp, e-mail, etc.). While the contents of these non-STOW packets is not known, the quantity of packets is known. If a particular simulation is over-supplying packets, the source will be easily identified. While such activity has the potential to cause problems during the exercise, it is felt that this is more of an operational/training issue and not primarily a monitoring issue, and should be dealt with accordingly.

The Work Station Load Monitoring supplies LAN managers with meaningful performance metrics for SAF and other Work Stations with minimal impact to network bandwidth and W/S CPU utilization. For ModSAF, simple CPU utilization is not a good performance metric because ModSAF runs at 100% CPU utilization even when idle. Frame rate is a better performance metric for ModSAF stations, as they are designed to run at a maximum frame rate of 15 Hz, and still operate effectively down to 3 or 4 Hz. The actual frame rate of any individual ModSAF station is an effective metric of how loaded the station is. For workstations other than ModSAF, CPU utilization measurements can be taken using tools that already exist within UNIX operating systems.

4.1 Monitoring Alternatives

Four main categories for monitoring alternatives have been researched and evaluated; Passive (listen only), Simple Network Management Protocol (SNMP), Remote Monitoring (RMON), and Simple Polling.

Passive network monitoring involves connecting a host (DEMvice) to a strategic location on the network of interest and monitoring each packet on that network. This approach does not poll other devices on the network to gather information about network traffic. Instead, it gets the same information by listening to the network, dissecting each packet, and pulling out information needed to determine its source and destination. This option does not effect network bandwidth (except when transferring information to DEM Central), but requires significant computing power to keep up with network traffic.

SNMP (e.g. HP OpenView, Sun Solstice, etc.) is a polling solution using the SNMP standard. These types of systems have been in existence for a number of years, are very stable, and have achieved wide acceptance and use in network monitoring. They also offer advanced graphics, user interfaces, and control functions already included in the package which do not need additional software development efforts. These systems basically poll other devices on the network to obtain required information. Some network bandwidth is required to acquire this information, the amount being a function of how often the system requests information. The more often the system asks for information, the more network bandwidth is required.

RMON and SNMP are industry standards for managing remote devices and LAN segments. SNMP can provide significant information about a device through polling of its public and private MIBs. However, SNMP does not support proactive monitoring and remote devices consume significant bandwidth when regularly polled from a central site. RMON was developed to support proactive monitoring of LAN traffic, but not the devices themselves. Using a combination of RMON and SNMP, it is possible to monitor LAN segment traffic and remote devices with minimal bandwidth impact.

Simple Polling uses either SNMP or some other polling method to obtain information from other devices on the network in the same manner a COTS SNMP system would. This approach requires significant software development and the construction of user interfaces.

5.0 Summary

This paper has described the required functions of the DISECT Distributed Exercise Manager (DEM) and the associated technical rationale for the decisions made. The DEM offers the capability to supply exercise control and monitoring functionality for large High Level Architecture (HLA) based simulation exercises, such as that planned for STOW 97. Exercise monitoring functions provide an efficient method of monitoring the performance of the hardware and software infrastructure of the simulation exercise and of detecting and isolating any problems that arise during the exercise. Exercise control functions allow an exercise manager to start, stop, and otherwise direct the exercise.

The DEM provides four main functions; Exercise Control, RTI Monitoring, Network Monitoring, and Work Station (W/S) Load Monitoring. The DEM system is designed to provide a centralized simulation exercise monitoring and control capability, referred to as DEM Central, and also provide for distributed monitoring within the STOW 97 network architecture using distributed platforms located at each LAN or subnet, referred to as DEMvices. Many of the functions of DEM cannot be performed without corresponding capabilities and services being provided by the RTI. RTI monitoring and exercise monitoring require that the RTI implemented for STOW 97 provide the capabilities described in the associated sections above. Network and workstation load monitoring are being implemented. The version of ModSAF being used in the STOW 97 exercise has been modified to incorporate the capability for DEM to accurately monitor the load of the ModSAF stations. A method exists using standard UNIX calls to provide at least some level of monitoring of systems that are not ModSAF based (e.g., environment servers, C4I systems). Several approaches to the task of network monitoring have been evaluated.